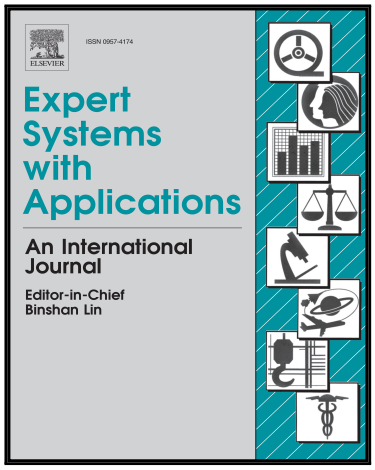
Accepted Manuscript



Accurate and Efficient 3D Hand Pose Regression for Robot Hand Teleoperation using a Monocular RGB Camera

Francisco Gomez-Donoso, Sergio Orts-Escolano, Miguel Cazorla

PII: S0957-4174(19)30463-4
DOI: https://doi.org/10.1016/j.eswa.2019.06.055
Reference: ESWA 12767

To appear in: *Expert Systems With Applications*

Received date: 12 November 2018
Revised date: 25 April 2019
Accepted date: 24 June 2019

Please cite this article as: Francisco Gomez-Donoso, Sergio Orts-Escolano, Miguel Cazorla, Accurate and Efficient 3D Hand Pose Regression for Robot Hand Teleoperation using a Monocular RGB Camera, *Expert Systems With Applications* (2019), doi: https://doi.org/10.1016/j.eswa.2019.06.055

Highlights

- A large-scale multi-view dataset that provides accurate annotations for hand poses
- A pipeline that improves the state-of-the-art results for 3D hand pose estimation
- We successfully applied our approach to robot hand teleoperation

Accurate and Efficient 3D Hand Pose Regression for Robot Hand Teleoperation using a Monocular RGB Camera

Francisco Gomez-Donoso[a,*], Sergio Orts-Escolano[a], Miguel Cazorla[a]

[a]*Institute for Computer Research, University of Alicante, Alicante, Spain.*

Abstract

In this paper, we present a novel deep learning-based architecture, which is under the scope of expert and intelligent systems, to perform accurate real-time tridimensional hand pose estimation using a single RGB frame as an input, so there is no need to use multiple cameras or points of view, or RGB-D devices. The proposed pipeline is composed of two convolutional neural network architectures. The first one is in charge of detecting the hand in the image. The second one is able to accurately infer the tridimensional position of the joints retrieving, thus, the full hand pose. To do this, we captured our own large-scale dataset composed of images of hands and the corresponding 3D joints annotations.

The proposal achieved a 3D hand pose mean error of below 5 millimeters on both the proposed dataset and Stereo Hand Pose Tracking Benchmark, which is a public dataset. Our method also outperforms the state-of-the-art methods.

We also demonstrate in this paper the application of the proposal to perform a robotic hand teleoperation with high success.

Keywords: Hand Pose Estimation, Deep Learning, Robot Teleoperation, Monocular

1. Introduction

Estimating the pose of an object or a human being is a highly challenging problem in Computer Vision. It is relevant for many robotics applications, ranging from robot

*Corresponding author

Email addresses: fgomez@ua.es (Francisco Gomez-Donoso), sorts@ua.es (Sergio Orts-Escolano), miguel.cazorla@ua.es (Miguel Cazorla)

teleoperation, human robot interaction (HRI), dexterous manipulation, navigation, etc. Besides, it is also relevant for other problems, such as activity recognition, Human-Computer Interaction (HCI), Virtual and Augmented Reality (VR/AR) and others. In the last decade, many works have addressed this problem, yet none of them solved it successfully from a computer vision perspective. In addition, estimating the pose of an object with multiple degrees of freedom, such as a human hand, presents several challenging issues that remain unsolved. In particular, dealing with the hand pose estimation problem adds additional complexities such as self-similarity and self-occlusion. The problem is even more difficult when dealing with hand-object interactions.

The recent advent of consumer depth cameras has resulted in the development of several approaches (Oikonomidis et al., 2011; Sridhar et al., 2015; Tagliasacchi et al., 2015; Tang et al., 2014) using an RGB-D camera to solve this problem. In particular, in recent years, we have also seen that the emergence of Convolutional Neural Networks (CNNs) has triggered the development of various methods that combine both RGB-D sensors and CNNs for 3D hand pose estimation and tracking (Tompson et al., 2014; Oberweger et al., 2015; Sinha et al., 2016; Mueller et al., 2017). These approaches have now been proven to work well for particular scenarios, such as performing mid-air frontal gestures within short distances. However, depth sensing is not available in most of our everyday computing devices, such as smartphones or laptops, limiting its availability to few users and applications. Being able to track and estimate 3D hand poses in real-world scenes using a commodity color camera remains an open problem. Deep learning-based techniques seem a promising direction for developing systems capable of tackling this problem. These techniques may have the capacity to learn models from RGB data that are able to deal with the full 3D hand/object pose estimation problem, including difficult scenarios with heavy occlusions caused by hand-object interactions.

In this work, we present a novel approach for real-time and accurate 3D hand pose estimation using a monocular RGB camera. It is a deep learning-based system capable of learning to accurately localize and estimate 3D hand poses. First, a region convolutional neural network is used to detect the hands in the input image. Then, the patch of the hand is forwarded to a convolutional neural network which is able to perform the regression of the 3D position of the joints in the hand.

Both networks are intelligent systems that are trained on our custom dataset. We demonstrate and validate the reliability and accuracy of the proposal on various public datasets. In addition, we propose the utilization of our intelligent system to control a robotic hand.

More specifically, our contribution in this work is three-fold:

- A novel large-scale multi-view dataset that provides ground truth annotation for multiple hand poses. Poses are captured from multiple viewpoints in a variety of scenarios: clutter, occlusions, egocentric viewpoint, different skin colors, etc, enabling the development of learning-based methods.
- A simple, yet effective, approach based on two-consecutive CNNs that improves state-of-the-art results for 3D hand pose estimation using a single, consumer RGB camera.
- We demonstrated the accuracy of the proposed method on a real robotics application: teleoperation system where the robot hand replicates human hand fine movements (fingers).

The proposed architecture significantly outperforms the state-of-the-art method recently presented in (Zimmermann & Brox, 2017), both in terms of accuracy and computational runtime.

2. Related Works

Several works have addressed the problem of 3D hand pose estimation. Although we can find related works from the early nineties, it is still an active problem for the computer vision community and highly challenging one. The first works on hand pose estimation relied on a traditional computer vision pipeline using traditional machine learning techniques. For example, the approach presented by (Stenger et al., 2004) used a cascade of classifiers arranged in a hierarchical structure in order to recognize multiple object classes. A shape-based descriptor is extracted in order to train the classifiers. These traditional approaches have some difficulties in estimating accurate hand poses. Besides, these methods lack generalization capabilities for new scenarios

and are computationally expensive. A more comprehensive review of earlier methods is presented in (Erol et al., 2007). In this section, we will review more recent works.

3D hand pose estimation from RGB-D sensors. With the advent of low-cost depth cameras most recent approaches utilized these 3D sensors. Researchers have addressed the hand pose estimation problem using information from both depth and RGB images. Within depth-based techniques, we find two different approaches. The first is based on generative, 3D model tracking (de La Gorce et al., 2011; Oikonomidis et al., 2011). A synthetic 3D hand model is normally used for generating hypotheses which are evaluated against real 3D data. This is noisy and has self-occlusions. This approach requires a precise poser initializer and is highly sensitive to large motions between frames, which is a common situation for human hands. The second approach using depth information is based on discriminative techniques. Works based on this approach mainly predict hand pose directly from RGB-D images. For example, (Kuznetsova et al., 2013) uses Random Forests for the prediction. (Tang et al., 2013) uses Random Forests for pixel-wise hand part classification utilizing as input depth images. In this way, they are able to detect hand parts, and finally, based on this estimation, the proposed system predicts 2D joint locations. This approach is computationally expensive due to the use of per-pixel classification techniques and also requires a huge training dataset to solve viewpoint discretization. In recent years, with the boom in Convolutional Neural Networks (CNNs) and deep learning techniques, new approaches for hand pose estimation have emerged. Most of these new techniques leverage CNNs for automatic feature extraction from depth images (Tompson et al., 2014). Other works combine automatic feature extraction with a supervised training to predict the 3D hand pose (Oberweger et al., 2015). These systems require a large of amount of labeled data (3D hand joints) for training.

The approach presented in (Sinha et al., 2016) is based on the utilization of a deep learning architecture, trained on synthetic data, to extract deep features. These features are organized according to their spatial and temporal neighborhood, achieving a robust and relatively jittering-free 3D hand pose estimation. This approach also works on depth maps and expects that the area of the image where the hand is located is previously detected. Another deep learning approach is described in (Tompson et al., 2014).

Their system uses a Convolutional Neural Network to compute a 3D hand pose that is later refined using predefined synthetic hand poses, namely, an inverse kinematics test. This work introduces an interesting development: the inclusion of the physical restrictions of the hand in the model. Although most of these methods start to work well for mid-air hand pose estimation, they all require the use of depth information obtained using a 3D sensor conveniently located within a short distance.

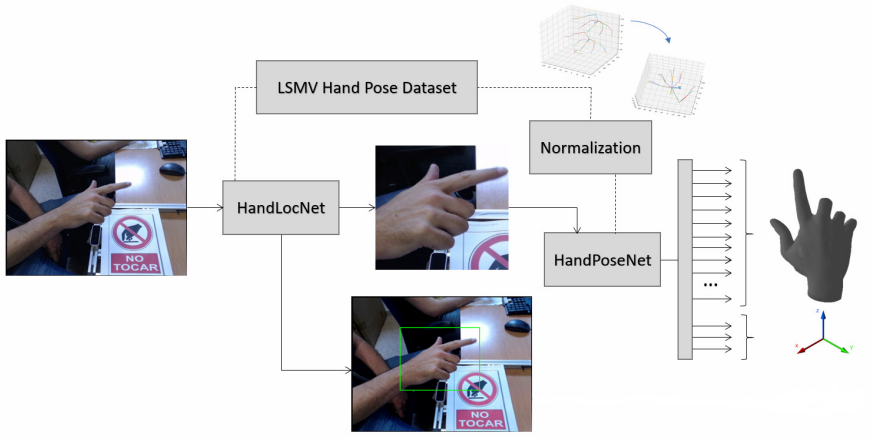


Figure 1: Pipeline overview: HandLocNet predicts the hand location. Then, the cropped hand image is forwarded to HandPoseNet, which performs the regression of the x,y,z positions for each hand joint. Additionally, it also infers the hand's orientation. The full pipeline is trained using our novel large-scale dataset.

3D hand pose estimation from single RGB images. (Zimmermann & Brox, 2017) is the only work that tackles the problem of 3D hand pose estimation using a single RGB image and CNNs so far. Three different CNNs architectures are used in this approach. The first one performs the user's hand segmentation at pixel level, the second detects 2D joints on the image space and, finally, the third CNN is trained to lift the detected keypoints to a 3D coordinate space. It is worth noting that this approach is fully trained using synthetic data and therefore does not provide accurate results for real data. Moreover, the use of three independent CNNs, including a network for hand segmentation (dense predictions), limit its performance.

Recently, other works have been published that tackle the 3D hand pose estimation

problem by using multi-view RGB cameras (Simon et al., 2017; Panteleris & Argyros, 2017). However, these methods are sensitive to camera calibration procedures and require prior depth estimation or at least, 3D triangulation across multiple RGB cameras (Simon et al., 2017).

Related datasets. A key element in deep learning-based systems is the large amount of data required for training. Some datasets for hand pose estimation use RGB-D data: Fully aligned RGB images, however, are often not provided (Tompson et al., 2014). Other existing datasets such as *Dexter* (Sridhar et al., 2016), provide incomplete hand annotation for our problem. The Stereo Hand Pose Tracking Dataset (Zhang et al., 2016) is one of the few that provides 2D and 3D annotation for all 21 hand joints, providing a total of $18,000$ stereo images. Two new large-scale datasets have recently been presented (Yuan et al., 2017; Garcia-Hernando et al., 2017). However, due to the capture system used for automatic annotation (six 6D magnetic sensors), the RGB images show sensors overlaid on top of user's hands, making it impossible to use these images for training a model that solely relies on RGB features. Some works have generated huge amounts of synthetic data (Mueller et al., 2017; Zimmermann & Brox, 2017) since this is fast and computationally inexpensive, but they lack generalization capabilities when dealing with real data.

As the revision of the state-of-the-art remarked, there exist two main different approaches to tackle the 3D hand pose estimation problem regarding the input data. First, there are methods that take as an input RGB-D frames. One main drawback of these approaches is that they need to deal with tridimensional data, which require high amount of both computation power and time. In addition, these approaches also demand the utilization of a close range depth sensor, such a Kinect or Intel SR300. These kind of cameras tend to not to perform well on outdoors or when the user's hand is relatively far from the sensor, rendering these methods unusable in outdoor environments. The other main approach to perform 3D hand pose estimation takes as input a single RGB frame. In this regard, the methods of the state-of-the-art are trained on synthetic data due to the lack of high amounts of annotated ground truth, which do not scale very well to real data.

Regarding the datasets, the reviewed works also show a huge gap for improvement. The considered datasets yield unaligned RGB and ground truth data. Anyway, in both cases the amount of data is insufficient to train deep learning-based models. There also exists other large-scale datasets with accurate ground truth, but they were captured using gloves and other intrusive devices that do not allow a natural depiction of the hands.

Given this review of the state-of-the-art we can conclude that the most convenient way to perform 3D hand pose estimation is to take as an input a single RGB frame captured by a single regular camera. It also can be concluded that there is a need for huge amounts of annotated ground truth. As for the datasets, those which provide images and tridimensional annotations of the joints do not provide enough data to train a robust system. This is mainly due to the difficulty of generating annotations for the images of real hands.

Thus, in this work, we introduce the following main novelties: First, a simple yet effective deep learning-based method for accurate hand localization and 3D pose estimation that takes a input a single RGB image from a regular camera. Then, a multi-view setup that allows us to create a large scale dataset of natural images of hands and the corresponding annotations in the 3D space. As our system is trained on this data, it is expected to work properly on real data. Finally, an application for the method in the robot teleoperation context is also introduced.

Finally, it is worth noting that our method for 3D hand pose estimation outperforms state-of-the-art results obtained by (Zimmermann & Brox, 2017) on public datasets.

3. 3D Hand Pose Regression

The main goal of this work is to perform 3D hand pose estimation using a single color camera. We used consumer web cameras, all providing images of $640x480$ pixels at 30Hz. Our pipeline is mainly formed by two different CNNs. The first, which we call $HandLocNet$ is trained to predict hand localization. It provides an accurate

cropped hand image that is used as an input for the second stage of the pipeline. Then, the second CNN, which we call HandPoseNet, performs 3D joint regression, inferring normalized 3D joint coordinates from the cropped RGB image. Both networks were trained using our novel large-scale dataset containing real data from multiple viewpoints. Figure 1 shows an overview of the proposed pipeline.

3.1. Hand model representation

In this work, the hand model is represented by two components. First, we define the hand pose using the 3D coordinates of its joints J. We define each joint position, $j_i = (x_i, y_i, z_i)$, for each hand part (metacarpal, proximal, intermediate and distal) of each finger (thumb, index, middle, ring and pinky). As in other works (Sridhar et al., 2015; Zimmermann & Brox, 2017; Qian et al., 2014), we used a total of 21 joints ($J = 21$). Moreover, to disambiguate hand orientation we also considered the orientation of the hand relative to the camera viewpoint. Thus, our representation is composed of 21 3D keypoints and an orientation vector $(\alpha_x, \alpha_y, \alpha_z)$, as shown in Figure 2.

Generally, 3D hand joint positions provided in existing datasets are given in global coordinates. This not only represents the hand pose but also the global position of the hand. In this way, two images are likely to be found on the dataset showing the same pose but which are translated with respect to each other. As our approach works on the hand cropped image, this could represent the very same pose, but ground truth data is different. Since the aim of this approach is to first infer the local hand pose, we applied a normalization process to the input data.

The normalization process consists on the creation of a hand local coordinate frame (root joint), so hand joints are transformed to the new coordinate frame, achieving translation and rotation invariance (see Equations 1). The new local coordinate frame is constructed by taking the normal vector of the hand (Z_e) and the vector (Y_e) going from the palm center (P_p) to the proximal knuckle of the index finger (P_{im}). Then, applying cross product we obtain (X_e). These three vectors represent an orthonormal system whose origin is located in the center of the palm (we used the center of the palm as the origin of the local coordinate frame since it is a stable keypoint). Then, we construct a transformation matrix (T_{norm}) from global to local coordinates using

the computed rotation matrix (R_{norm}) and translation vector t_{norm}. Formally, we will refer to the normalized 3D joints as $j_i^{norm} \in \mathbb{R}^{3n}$.

$$Y_e = P_p - P_{im} \tag{1a}$$

$$X_e = Ze \times Y_e \tag{1b}$$

$$R_{norm} = [X_e, Y_e, Z_e] \tag{1c}$$

$$t_{norm} = -R_{norm}^T * P_p \tag{1d}$$

$$T_{norm} = [R|t_{norm}] \tag{1e}$$

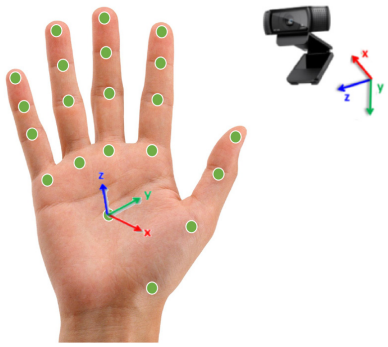
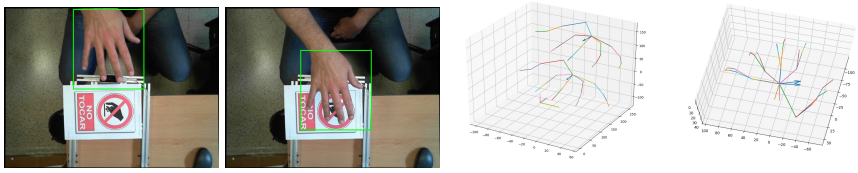


Figure 2: Our hand representation model consist of 21 keypoints corresponding to the joints of each finger and the hand orientation relative to the camera viewpoint ($\alpha_x, \alpha_y, \alpha_z$).

Figure 3 shows the representation of two dataset samples before and after the normalization process.

3.2. HandLocNet: Hand Localization Net

The proposed hand localization network (HandLocNet) is based on the work originally presented in (Redmon & Farhadi, 2017). In contrast to other related works that tackle this problem using segmentation (Mueller et al., 2017) and encoder-decoder (probability heatmap) networks (Zimmermann & Brox, 2017), we cast this problem as an object localization (box prediction) task (Ren et al., 2015; Redmon & Farhadi,



(a) (b) (c) (d)

Figure 3: Normalization process for two similar hand poses (a,b,c). After the normalization, the 3D joint positions on their local coordinates are almost identical (d).

2017). Our HandLocNet is a shallower version of the network from (Redmon & Farhadi, 2017), achieving a good balance between runtime and accuracy. The proposed hand detector is able to localize hands in challenging scenarios: clutter, occlusions, novel viewpoints, skin color, object-interaction, etc. We trained $HandLocNet$ using a new large-scale multi-view hand pose dataset (Section 4). We used pre-trained weights from the PASCAL VOC dataset (Everingham et al., 2015). HandLocNet predicts the hand location and a confidence score σ, and so predictions with a low confidence score are discarded. We empirically set $\sigma = 0.85$. It is worth noting that HandLocNet is able to accurately detect hands even when they are holding or manipulating objects, as shown in Figure 4.

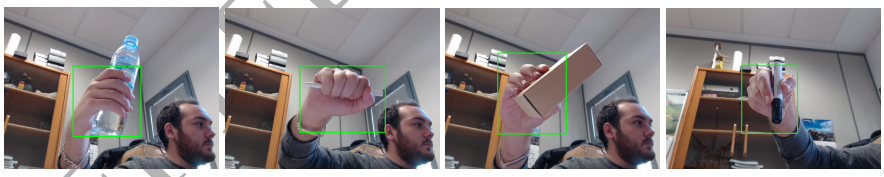


Figure 4: HandLocNet is able to accurately detect hands even when manipulating objects

3.3. 3D hand joints regression

In our approach, we cast the 3D hand pose estimation problem as a regression task. The proposed HandPoseNet architecture takes as an input RGB images of 224×224 pixels. The cropped hand image predicted is then resized and passed as input data. The hand pose is estimated by HandPoseNet. This architecture is derived from

ResNet50 (He et al., 2015), and has been modified to regress normalized 3D hand joints j_i^{norm} given the detected hand subimage. Our normalized representation (palm-relative positions) renders the regression of hand 3D joints invariant to translation and rotation. We tested direct regression but, given the complexity of the problem, we found that the network was overfitting to the training data. The number of neurons for the last fully connected layer is modified to estimate hand joints, each of them formed by its x, y and z coordinates. Additionally, the network estimates the hand orientation, α_x, α_y and α_z. HandPoseNet is trained using the new large-scale multi-view dataset presented in Section 4.

4. Large-Scale Multi-View Hand Pose Dataset

This works presents a novel Large-Scale Multi-View Hand Pose dataset, which we call *LSMVHandPoses*. We have developed a framework for semi-automatic ground truth annotation using the Leap Motion sensor (LeapMotion, 2017) and multiple RGB cameras. The proposed dataset is intended to be used for 3D and 2D hand pose estimation, including 2D hand localization. To generate accurate ground truth annotations, a custom multi-camera rig was created, as shown in Figure 5. This device consists of an aluminum structure with three articulated arms holding a total of four cameras. We calibrated each camera individually, computing intrinsic parameters K for each sensor, including the hand tracker. Finally, we also computed extrinsic parameters (Zhang, 2000), roto-translation matrices, between all cameras and the Leap Motion sensor (LeapMotion, 2017). The proposed dataset and supplementary material is available at our project website[1] `http://www.rovit.ua.es/dataset/mhpdataset/`.

The dataset is organized in various sequences and each sequence is composed of a set of frames. A frame contains the following information:

- 3D hand joints as provided by the Leap Motion sensor (raw data).
- Four RGB images from four different perspectives (top, bottom, left and right).

[1] `http://www.rovit.ua.es/dataset/mhpdataset/`

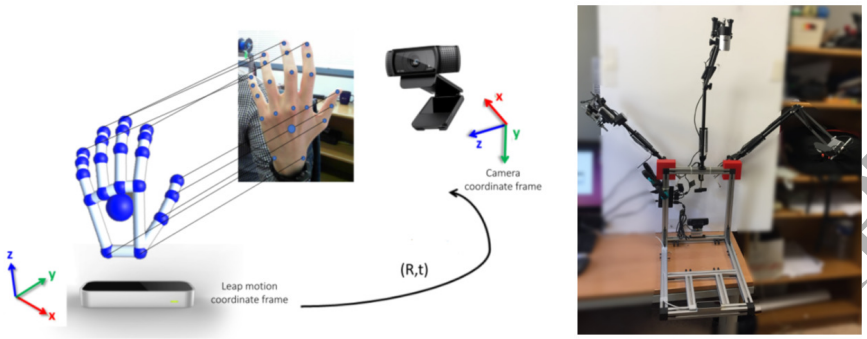


Figure 5: We created a custom structure (multi-camera rig) in order to generate the dataset (right). It allows us to quickly generate large amounts of ground truth data. We performed a calibration process (intrinsics and extrinsics) to generate 3D hand pose annotations for each color camera (left).

- Four sets of 2D points as the resultant projection of the 3D points to each RGB camera coordinate frame.
- Four bounding boxes computed from the projection of the 3D points to each camera coordinate frame.

Finally, it is worth noting that each sequence was captured in different conditions to assure high variability. Subject, moment of the day, motion speed, hand-object interaction, skin color and light conditions are some of the parameters we considered to achieve high variability.

This dataset contains over $20,500$ different frames distributed in 21 sequences. For each frame, 4 color images and 9 different annotations were provided. Thus, over $80,000$ color images and over $184,500$ annotations in total are provided. Figure 6 shows a random sample of the dataset and its ground truth.

4.1. Ground-truth data accuracy

We can find that the accuracy of the Leap Motion has been exhaustively explored in (Guna et al., 2014; Weichert et al., 2013), where they concluded that the mean error is about $4mm$ whilst in the worst cases is under $9mm$. Nonetheless, Figure 7 shows some frames extracted from our dataset with the 3D joints estimated by the Leap Motion sensor and reprojected back to 2D (green points) for qualitative analysis.

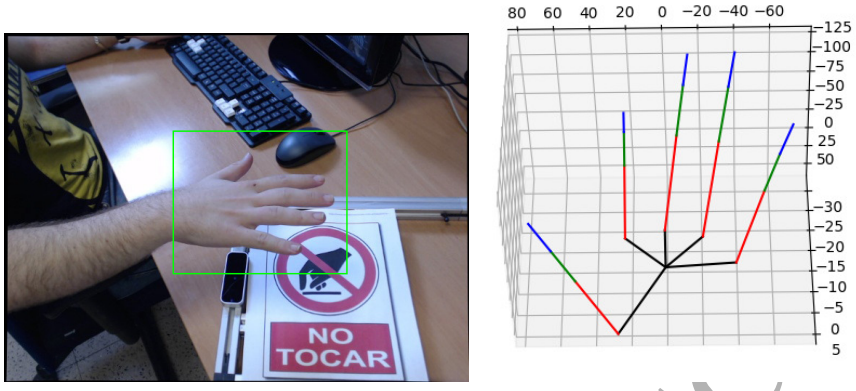


Figure 6: A random sample extracted from the LSMVHandPoses dataset, RGB image, hand bounding box and 3D joint locations (ground truth).

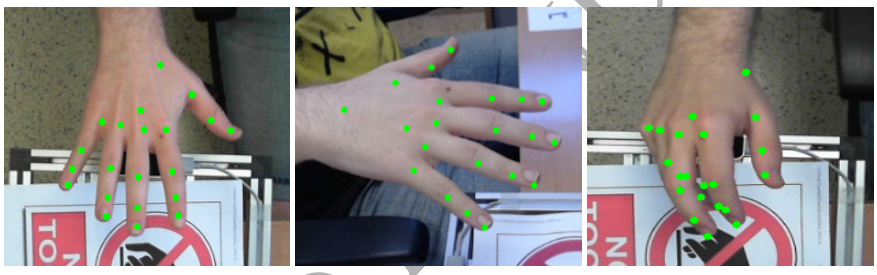


Figure 7: Random images from the dataset that show the reprojected 2D annotations from the Leap Motion sensor (green) and (user) manually-annotated hand joints (red).

Additionally, in order to evaluate the ground-truth precision of the generated data, we manually annotated 100 frames and computed the mean distance error between the manually annotated 2D joints, and the ones estimated by reprojecting the 3D joints provided by the Leap Motion sensor. Using the set of manually annotated frames we obtained a mean error of 12.2 pixels, which is less than 2% of the horizontal image resolution.

5. Evaluation and results

The subsequent sections describe in more detail the experiments we performed to validate our approach. We also present details of the training procedure. As previously

stated, our pipeline is composed of two main trainable parts. First, HandLocNet computes the localization of the hand (bounding box) and then, HandPoseNet performs 3D joint regression. Both systems were trained on the new *LSMVHandPoses* dataset.

5.1. Benchmarks

In addition to evaluating our approach on the dataset presented in section 4, we have also evaluated it on the following public datasets:

Stereo Hand Pose Tracking Benchmark (Zhang et al., 2016) This dataset comprises 12 videos: 6 depict different individuals counting with their hands. The other 6 consist of random unconstrained hand poses. The background is inconsistent across the samples in order to introduce high variability. The sequences were recorded using three sensors: a stereo pair and a depth camera. As our approach uses RGB images, the depth images are discarded. In total, this dataset contains $54k$ samples.

Rendered Hand Pose Dataset (Zimmermann & Brox, 2017) This dataset is composed of a variety of synthetic images, rendering 20 characters performing 39 different actions. Each frame is generated using a random camera pose and a random background extracted from a pool of real landscapes. They also generate random lighting directions and intensities in order to ensure high variability.

In total, this dataset provides $41,258$ images for training and $2,728$ for testing. The image resolution is $320x320$ pixels. The annotations are composed of x,y and z positions for the 21 hand joints. In addition they provide depth maps and 2D hand keypoints (camera image plane).

This dataset provides precise annotations but also has a severe disadvantage: the rendered images are not sufficiently realistic. As the samples differ greatly from reality, training a system only on this dataset is not feasible. Nonetheless, this dataset could be used to slightly improve the generalization capabilities when jointly trained with another realistic dataset such as the one presented in Section 4.

5.2. Network training and inference runtime

The HandLocNet (hand location) model was pre-trained using the PASCAL VOC dataset (Everingham et al., 2015) and then fine-tuned using the novel *LSMVHand-*

Pose dataset. We used the loss function defined in (Redmon & Farhadi, 2017). The HandPoseNet (3D joint regression) is derived from the ResNet50 architecture. It is trained using the proposed dataset, but original ResNet50 layers were initialized with the weights from the trained *ImageNet* model. Other layers weights were randomly initialized. HandPoseNet uses a single squared error loss term, which is suitable for regression problems. Specifically, we have used the mean square error of all components of our hand model. We used the tensorflow framework and the Adam solver for training.

The systems runs live at 35 frames per second. The whole pipeline leverages GPU processing power, being fully implemented on the GPU. The proposed system has been tested on a NVIDIA GTX 1080Ti. HandPoseNet (3D joint regression) takes on average 17.6 milliseconds and the HandLocNet (hand localization) takes on average 10.1 milliseconds.

5.3. Results: LSMVHandPose dataset

In this experiment, both CNNs were trained on the *LSMVHandPose* dataset, which was split in training and test sets (80% and 20% ratio, respectively). HandLocNet was trained using ground truth bounding boxes obtained using the proposed framework for automatic hand pose annotation (Section 4). It achieves a 0.89 Intersection over Union score on the test set, providing accurate predictions for hand localization.

The HandPoseNet CNN was trained on a preprocessed version of the ground truth data (normalization), as explained in Section 3.1. The regression error was computed by comparing the ground truth joint positions with the network output using a mean squared error term. Finally, the computed errors for the estimated hand joints are aggregated and averaged.

This experiment produced a mean error of 4.1556 millimeters on the test set, proving its high accuracy. In fact, 74.80% of the test samples yield a mean error under 5 millimeters. Moreover, as in (Zimmermann & Brox, 2017), we report the area under the curve on the Percentage of Correct Keypoints (PCK), Figure 8.

As shown in Figure 9, the system is able to accurately infer the 3D hand joints and the orientation of the hand relative to the camera viewpoint. Apart from the samples

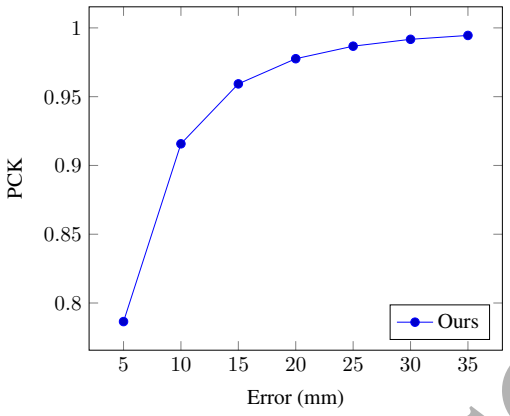


Figure 8: Results for 3D joint regression on the test set of the LSMVHandPose dataset. It shows the Percentage of Correct Keypoints (PCK) over the corresponding thresholds in millimeters.

Figure 9: From left to right: cropped hand image, ground truth and predicted 3D joint positions using our approach (ground truth is shown in red whilst inferred positions are depicted in green), regressed hand pose (rigged hand model).

that yield high error due to sensor problems such as blurry images, there are other factors that harm the accuracy of the system. It still can be observed that eventualities such as self-occlusions or lack of examples of certain poses negatively impact on the accuracy of the predictions.

5.4. Results: Stereo Hand Pose Tracking Benchmark and Rendered Hand Pose Dataset

In this experiment, we trained the proposed HandPoseNet network on the Stereo Hand Pose Tracking Benchmark and the Rendered Hand Pose Dataset, mirroring the experiments presented in (Zimmermann & Brox, 2017). In this case, the 3D infor-

mation was normalized following the procedure shown in Section 3.1. The error was computed in the same way explained earlier in Section 5.3.

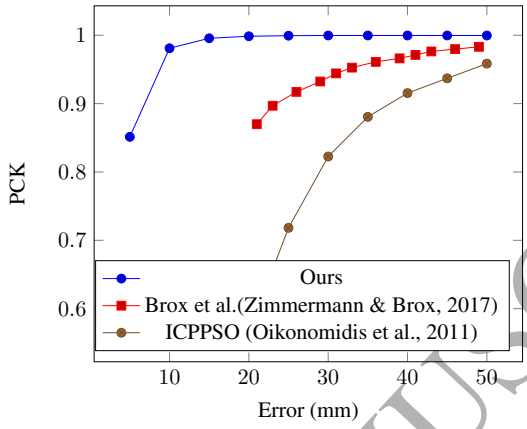


Figure 10: 3D joint regression results on the test split of the Stereo Hand Pose Tracking Benchmark. It shows the percentage of correct keypoints (PCK) over the corresponding thresholds in millimeters.

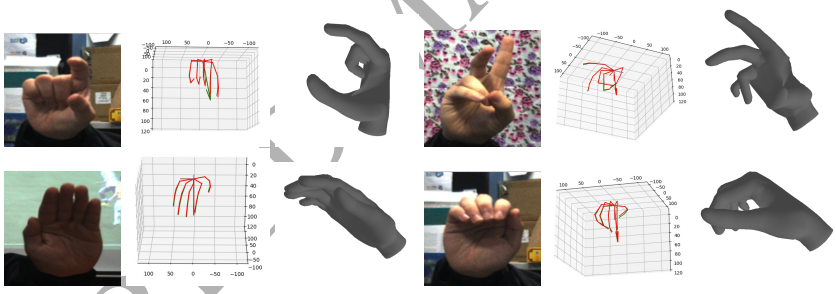


Figure 11: From left to right: cropped hand image, ground truth and predicted 3D joint positions using our approach (ground truth is shown in red whilst inferred positions are depicted in green), regressed hand pose (rigged hand model).

Figure 11 depicts estimated hand poses and their ground truth annotations for some samples in the dataset. As shown in Figure 11, ground truth and inferred hand poses almost fully overlap due to the high accuracy of the approach. Evaluating our approach on the test split we obtained an error of 4.14038 millimeters. In fact, 76.67% of the test samples yield a mean error below 5 millimeters. Figure 10 shows the accuracy of the system depending on the error threshold. In this figure, as in (Zimmermann &

Brox, 2017), we plot the area under the curve on the Percentage of Correct Keypoints (PCK). This experiment allows a straightforward comparison with (Zimmermann & Brox, 2017). In that work, described in Section 2, the authors performed a similar experiment with the same public datasets. As stated in their work, they achieved < 0.9 PCK for a 21 millimeters threshold. Under the same circumstances, our approach achieves 0.9985 PCK. Moreover, our approach achieves 0.9 PCK between 5 and 10 millimeters thresholds. We cannot further compare both methods on lower thresholds as the authors only reported results from 20 millimeters error onwards.

5.5. Comparison with the state of the art methods

We compare our results with two of the best performers in the state of the art for hand pose estimation. These methods are Zhang et al. (2016)(ICPPSO) and Zimmermann & Brox (2017) as mentioned before. As the Sections 5.3 and 5.4 stated, our method outperforms them both in terms of accuracy. Nonetheless, our method shows additional advantages. For instance, ICPPSO is based on stereo methods. It means that in order to create a good tridimensional representation, the stereo setup must undergo a complex calibration step. Nonetheless, our approach uses a single color image as an input that does not require a special setup or calibration. In addition, it is known that the stereo matching methods do not perform well on low texture scenarios. Finally, ICPPSO requires a training stage to learn the color skin tones and background textures to perform robustly. Due to this, it is expected an accuracy drop when fed with unknown scenarios. However, our HandLocNet is trained on real data and is able to generalize to unknown scenarios.

Regarding Brox et al., it is based on an ensemble of three independent different networks that requires a complex training process. On the contrary, our method relies on a two step ensemble that is trained with no special considerations and follows solid and tested approaches such a region detection and regression. In addition, their segmentation network follows a pixel-wise classification approach which tends to be more error prone than a region detection method. Furthermore,

this part of the ensemble is trained on synthetic data which limits the performance on real data. However, out system is fully trained on real data that is able to perform robustly in unknown scenarios.

6. Robot Hand Teleoperation

In the last years, robotics Artificial Intelligence (AI) has drastically changed enabling more complex behaviors and allowing them to perform tasks that were not possible before, such as 3D object recognition, 3D pose estimation, incremental learning, automatic localization and planning, and so on. However, we are still far from robots that are fully autonomous, and able to carry out any kind of tasks, like a human being. That being said, robot teleoperation is still a useful application that allows using robots to explore areas that may be dangerous for the human being, such as operating in industrial environments, or exploring a nuclear plant that has suffered an accident. Besides, it is common for humans to carry out fine actions like to tie a knot, inserting a pin in a hole or to use a hand tool (fine motor skills). They may seem like simple tasks, but are really complex and involve extremely fine finger and hand motions. Human hands are excellent at those tasks.

Most of the previous approaches for precise robot hand teleoperation require using intrusive gloves, such as (Fang et al., 2017; Park et al., 2017). Using an electronic glove has many potential problems, electro-magnetic interference, requires specific hardware, high cost, etc. Recently, some works have used RGB-D cameras, such as the Microsoft Kinect, for performing gesture recognition and hand tracking. 3D hand information provided from the range sensor is used to teleoperate a robot arm/hand in a coarse way, being limited to perform simple pick-and-place tasks (Du & Zhang, 2014).

Human robot interaction and robot teleoperation needs to be simple, natural and affordable, which is the reason why we proposed a natural way of interaction by replicating hand movements in front of a consumer webcam. The purpose of this research is to develop a robot hand system which is controlled with precise human hand motions, developing a teleoperation system where the robot hand replicates human hand motions. As an example application, we used the proposed system for controlling a

simulated humanoid robot hand (AR-10). Output 3D hand poses are used to replicate human hand motion in the RViz visualizer (ROS), so we generate a real-time simulation using a robot hand. The same system can be used to teleoperate other existing robotic hands, such as the ShadowHand (Sha, 2018). Figure 12 shows the replicated hand poses performed by a human in front of a regular webcam. It can be appreciated that the simulated robot hand is able to accurately replicate the human hand pose. Supplementary material shows some videos of the system working in real-time.

7. Conclusions

In this work, we have presented an approach for 3D hand pose estimation using a single RGB camera. This approach is able to accurately predict the 3D position of each joint in a hand, thus, retrieving a 3D hand pose relative to the camera viewpoint. Our method takes advantage of two CNNs to detect and regress, in real-time, the 3D joint positions of a hand. We present quantitative and qualitative experimentation on public datasets to validate the accuracy of our approach. Furthermore, comparison with other methods show that our method achieves higher accuracy than the state-of-the-art system presented in (Zimmermann & Brox, 2017). We can also conclude that the presented system is able to achieve very similar performance than a Leap Motion device but just using a single, low-cost, color camera. Moreover, we also presented *LSMVHandPoses*, a novel dataset that we used for training our approach. It provides ground truth annotation for multiple hand poses. These poses were captured from multiple viewpoints in a variety of scenarios: clutter, occlusions, egocentric viewpoint, different skin colors, etc.

Finally, we qualitatively evaluated the accuracy of the proposed method on a robotics application: robot teleoperation system where the humanoid robot hand replicates human hand fine movements.

We illustrated the applicability of our method within a teleoperation of robotic hands context, but the ability to estimate the tridimensional pose of the hand from a single RGB image has a range of purposes. For instance, our system could be

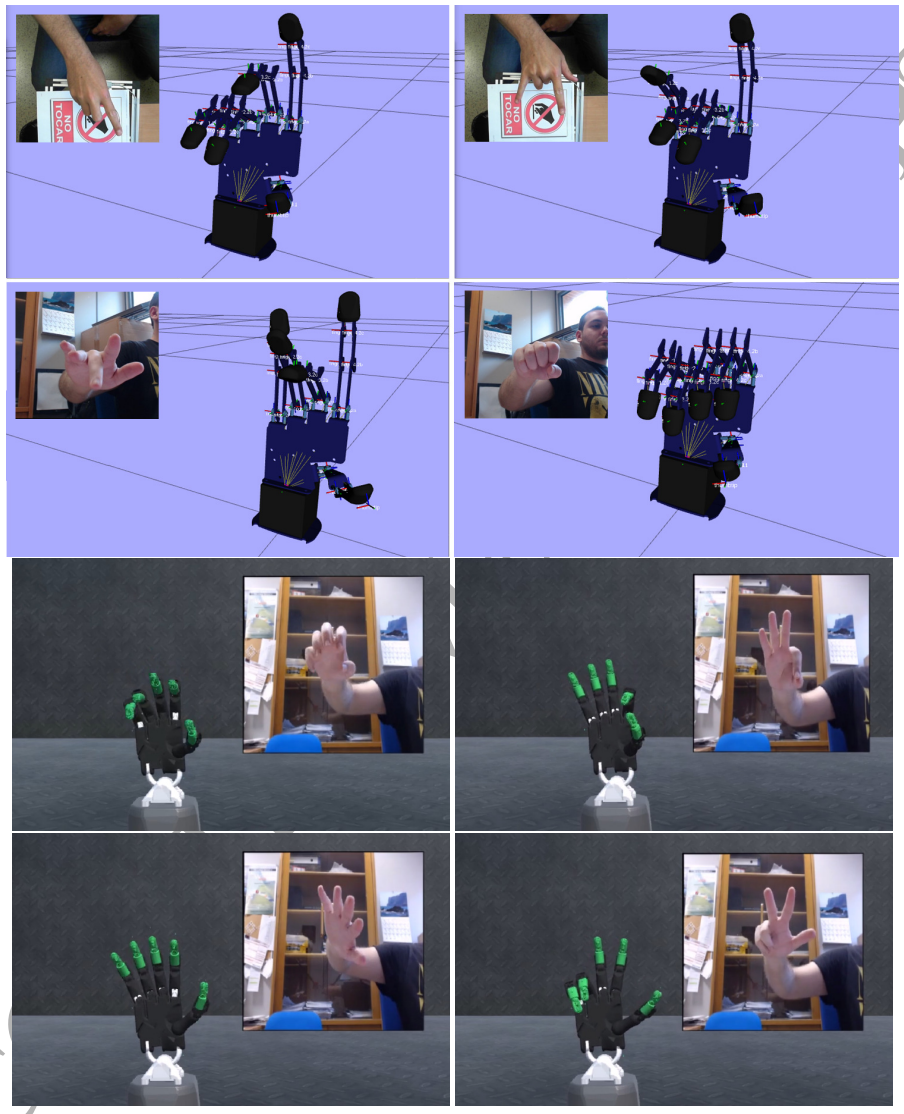


Figure 12: Simulation of an AR-10 and a Shandow Hand humanoid robot hands being teleoperated using our system (Human hand pose replication).

used as a prior step to perform sign language recognition. Our proposal could also enable new gesture-based human user interfaces to control computers, robots and other devices. It is also relevant in the virtual reality context as the users could use their hand in a natural fashion to interact with the elements within the virtual world.

It is worth noting, that our proposal would enable all these applications with no investment at all as it only requires a regular color camera. In addition, as it is non-invasive, it will not limit the user movements and will not decrease the immersion sensation of the user when applied to virtual reality.

7.1. Limitations

Despite the accuracy of the proposal, our system suffers from a minor jittering when the results are rendered over time. This effect arises when we predict the 3D pose of the hand of two subsequent frames. Due to the error of the predictions, which are frame-wise, the hand poses are not fully consequent over time, thus leading to a noticeable jittering effect. Another weakness of our hand pose estimation proposal is that it relies on two different deep learning-based architectures. Despite running in a real-time fashion, the ensemble is somewhat inefficient.

8. Future works

So in order to solve the mentioned issues, we plan to extend our method to smooth the results over time and propose and end to end approach. To do so, we intend to use recurrent units inside HandPoseNet. The recurrent units are able to take sequences as input and learn temporal patterns. We hope that this approach could help fighting against the jittering effect. In addition, a post process will be also considered to filter and smooth the predictions. In addition, we also plan to join HandLocNet and HandPoseNet in one architecture. This would hopefully enable a simpler training procedure and a faster inference step. To top that, as the proposed methodology to create the dataset allow an easy and fast capture of new ground truth, we also plan to extend the dataset by involving more individu-

als. This would improve the generalization capabilities of both HandLocNet and HandPoseNet models.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work has been supported by the Spanish Government TIN2016-76515R Grant, supported with Feder funds. This work has also been supported by a Spanish grant for PhD studies ACIF/2017/243. Thanks also to Nvidia for their generous donations.

The authors declare no competing financial interests.

References

References

(2018). Shadow dexterous hand. URL: `http://www.shadowrobot.com/products/dexterous-hand/`.

Du, G., & Zhang, P. (2014). Markerless human-robot interface for dual robot manipulators using kinect sensor. *Robot. Comput.-Integr. Manuf.*, *30*, 150–159.

Erol, A., Bebis, G., Nicolescu, M., Boyle, R. D., & Twombly, X. (2007). Vision-based hand pose estimation: A review. *Comput. Vis. Image Underst.*, *108*, 52–73.

Everingham, M., Eslami, S. M. A., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, *111*, 98–136.

Fang, B., Sun, F., Liu, H., & Guo, D. (2017). A novel data glove using inertial and magnetic sensors for motion capture and robotic arm-hand teleoperation. *Industrial Robot*, *44*, 155–165.

Garcia-Hernando, G., Yuan, S., Baek, S., & Kim, T. (2017). First-person hand action benchmark with RGB-D videos and 3d hand pose annotations. *CoRR*, .

Guna, J., Jakus, G., Pogačnik, M., Tomažič, S., & Sodnik, J. (2014). An analysis of the precision and reliability of the leap motion sensor and its suitability for static and dynamic tracking. *Sensors*, *14*, 3702–3720.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. *CoRR*, .

Kuznetsova, A., Leal-Taixé, L., & Rosenhahn, B. (2013). Real-time sign language recognition using a consumer depth camera. In *2013 IEEE International Conference on Computer Vision Workshops* (pp. 83–90).

de La Gorce, M., Fleet, D. J., & Paragios, N. (2011). Model-based 3d hand pose estimation from monocular video. *IEEE Trans. Pattern Anal. Mach. Intell.*, *33*, 1793–1805.

LeapMotion (2017). `https://developer.leapmotion.com/orion`, version 2.3.1.

Mueller, F., Mehta, D., Sotnychenko, O., Sridhar, S., Casas, D., & Theobalt, C. (2017). Real-time hand tracking under occlusion from an egocentric rgb-d sensor. In *The IEEE International Conference on Computer Vision (ICCV)*.

Oberweger, M., Wohlhart, P., & Lepetit, V. (2015). Training a feedback loop for hand pose estimation. In *2015 IEEE International Conference on Computer Vision (ICCV)* (pp. 3316–3324).

Oikonomidis, I., Kyriazis, N., & Argyros, A. A. (2011). Full dof tracking of a hand interacting with an object by modeling occlusions and physical constraints. In *ICCV* (pp. 2088–2095).

Panteleris, P., & Argyros, A. A. (2017). Back to rgb: 3d tracking of hands and hand-object interactions based on short-baseline stereo. In *IEEE International Conference on Computer Vision Workshops (HANDS 2017 - ICCVW 2017)*. (pp. 575–584). Venice, Italy.

Park, Y., Jo, I., Lee, J., & Bae, J. (2017). A wearable hand system for virtual reality. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 1052–1057).

Qian, C., Sun, X., Wei, Y., Tang, X., & Sun, J. (2014). Realtime and robust hand tracking from depth. In *2014 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1106–1113).

Redmon, J., & Farhadi, A. (2017). Yolo9000: Better, faster, stronger. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems 28* (pp. 91–99).

Simon, T., Joo, H., Matthews, I., & Sheikh, Y. (2017). Hand keypoint detection in single images using multiview bootstrapping. In *CVPR*.

Sinha, A., Choi, C., & Ramani, K. (2016). Deephand: Robust hand pose estimation by completing a matrix imputed with deep features. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Sridhar, S., Mueller, F., Oulasvirta, A., & Theobalt, C. (2015). Fast and robust hand tracking using detection-guided optimization. In *Proceedings of Computer Vision and Pattern Recognition (CVPR)*.

Sridhar, S., Mueller, F., Zollhoefer, M., Casas, D., Oulasvirta, A., & Theobalt, C. (2016). Real-time joint tracking of a hand manipulating an object from rgb-d input. In *Proceedings of European Conference on Computer Vision (ECCV)*.

Stenger, B., Thayananthan, A., Torr, P. H. S., & Cipolla, R. (2004). Hand pose estimation using hierarchical detection. In N. Sebe, M. Lew, & T. S. Huang (Eds.), *Computer Vision in Human-Computer Interaction: ECCV 2004 Workshop on HCI, Prague, Czech Republic, May 16, 2004. Proceedings*.

Tagliasacchi, A., Schröder, M., Tkach, A., Bouaziz, S., Botsch, M., & Pauly, M. (2015). Robust articulated-icp for real-time hand tracking. In *Proceedings of the Eurographics Symposium on Geometry Processing* (pp. 101–114).

Tang, D., Chang, H. J., Tejani, A., & Kim, T.-K. (2014). Latent regression forest: Structured estimation of 3d articulated hand posture. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 3786–3793).

Tang, D., Yu, T. H., & Kim, T. K. (2013). Real-time articulated hand pose estimation using semi-supervised transductive regression forests. In *2013 IEEE International Conference on Computer Vision* (pp. 3224–3231).

Tompson, J., Stein, M., Lecun, Y., & Perlin, K. (2014). Real-time continuous pose recovery of human hands using convolutional networks. *ACM Trans. Graph.*, *33*, 169:1–169:10.

Weichert, F., Bachmann, D., Rudak, B., & Fisseler, D. (2013). Analysis of the accuracy and robustness of the leap motion controller. *Sensors*, *13*, 6380–6393.

Yuan, S., Ye, Q., Garcia-Hernando, G., & Kim, T. (2017). The 2017 hands in the million challenge on 3d hand pose estimation. *CoRR*, .

Zhang, J., Jiao, J., Chen, M., Qu, L., Xu, X., & Yang, Q. (2016). 3D Hand Pose Tracking and Estimation Using Stereo Matching. *ArXiv*, .

Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, *22*, 1330–1334.

Zimmermann, C., & Brox, T. (2017). Learning to estimate 3d hand pose from single rgb images. In *The IEEE International Conference on Computer Vision (ICCV)*.

Author contribution

Miguel Cazorla and Sergio Orts: Conceptualization, Methodology, Supervision, Validation, Writing-Reviewing and Editing, Supervision. Francisco Gomez-Donoso: Software, Data curation, Writing-Original draft preparation, Visualization, Investigation.